Stjelja, Davor; Jokisalo, Juha; Kosonen, Risto

Futureproofing and scaling machine learning for occupancy prediction

Published in:
Proceedings CLIMA2022 : REHVA 14th HVAC Congress, 22-25 May 2022, Rotterdam

DOI:
10.34641/clima.2022.95

Published: 17/05/2022

Document Version
Publisher's PDF, also known as Version of record

Futureproofing and scaling machine learning for occupancy prediction

Davor Stjelja [a,b], *Juha* Jokisalo [a], *Risto* Kosonen [a,c]

[a] Department of Mechanical Engineering and Automation, Aalto University, 02150 Espoo, Finland; davor.stjelja@aalto.fi juha.jokisalo@aalto.fi; risto.kosonen@aalto.fi

[b] Innovations, Granlund Oy, 00701 Helsinki, Finland; davor.stjelja@granlund.fi

[c] College of Urban Construction, Nanjing Tech University, Nanjing 211800, China

Abstract. An important instrument for achieving smart and high-performance buildings is Machine Learning (ML). A lot of research was done in exploring the ML learning models for various applications in the built environment such as occupancy prediction. Nevertheless, this research focused mostly on analyzing the feasibility and performance of different supervised ML models but have rarely focused on practical applications and scalability of those models. In this study, we are proposing a transfer learning method as a solution to few typical problems with the practical application of ML in buildings. Such problems are scaling a model to another (different) building, collecting ground truth data necessary for training the supervised model and adapting the model when conditions change. The practical application examined in this work is a deep learning model used for predicting room occupancy using indoor air quality (IAQ) IoT sensors. The importance of occupancy prediction has risen in recent times of remote work and is especially important for futureproofing of the built environment. This work proves that it is possible to reduce significantly the need for ground truth data collection for deep learning based occupancy detection model. Additionally, the robustness of the transferred model was tested, where performance stayed on similar level if suitable normalization technique was used.

Keywords. Occupancy prediction, environmental sensors, deep learning, transfer learning, scalability, practical issues
DOI: https://doi.org/10.34641/clima.2022.95

1. Introduction

The CO_2 emissions of buildings in European Union are 36% and 28% on global scale [1], [2]. While HVAC systems in developed countries are responsible for 50% of building energy consumption alone [3]. To battle this problem, the EU has set the goal of developing a sustainable, competitive, secure and decarbonized energy system by the year 2050 [1].

One of the tools to achieve this goal is through the digitalization of energy systems and buildings and the EU has introduced a smart readiness indicator (SRI). The purpose of this indicator is to determine the capability of buildings in using information and communication technologies to adapt the building operation to the needs of the occupants and the grid while improving the overall performance of the buildings [1]. Recently, many research attempts have been made using advanced technologies in the field of computer science such as artificial intelligence (AI), machine learning (ML) and the internet of things (IoT) in building operations. Enabling use cases such as model predictive control, system fault detection and diagnosis, occupancy estimation and detection, demand response and in more general system integrator of different building subsystems and occupants [4], [5]. These technologies can help to significantly improve building performance, from higher indoor environment quality, lower energy consumption to better space efficiency.

Office building occupancy inefficiency is a significant potential for improvement, as in one British study from 2013 has concluded that regular offices had an average occupancy level between 60 and 70% [6]. The recent COVID-19 pandemic has fast tracked remote work making occupancy levels worse, where in one Israeli study more than 60% of interviewees expects to come to the office only two to three times a week after the end of pandemics [7]. If these claims come true, existing buildings and their HVAC systems can become even more inefficient, since the standard static occupancy profile, with which they were designed with, is not valid anymore.

This all leads to the expectation that information on building occupancy is going to become more valuable and needed than ever before as it is important for optimizing the space and building's system operation. Fortunately, a considerable amount of literature has been published on the matter of building occupancy estimation and detection [8].

To infer and analyze the information on occupancy in a larger scale, the commonly available sensors in buildings should be utilized. Environmental sensors commonly available in buildings are CO_2, room air temperature, relative humidity and in some case total volatile organic compound (TVOC) of which CO_2 has highest correlation with occupants in a room [9], [10].

For occupancy prediction from room environmental data methods range from physical, gray-box, statistical, machine learning to deep learning models. Where in this work focus was on machine and deep learning methods. Machine learning methods, have shown good accuracy with occupancy prediction, but require a process which is called feature engineering [11]–[15]. This time-consuming process was solved using deep learning methods such as in [16], where proposed method has outperformed statistical and machine learning methods. However, for using the model in a different room, the authors have needed to collect new ground truth data and train the model from scratch. The collection of occupancy ground truth is challenging and has been identified in several works as costly and time consuming [5], [17], [18].

In this work solving the problem of deep learning method scalability for room occupancy prediction is attempted. This is done using transfer learning, which is a technique for extracting knowledge from extensive and known source dataset and using it to improve the learning of a model on the lesser-known target dataset [19]. A similar work was done by Weber et al. [20], but using the simulated data of the same room as a source model. Here, a model trained from one room's data is transferred to another room in a different building, for which much fewer training data is available. Additionally, transferred model robustness is tested with different ventilation setting as well.

2. Methodology

2.1 Overview of CDLSTM model

Deep learning model used for predicting occupancy from environmental variables in this work is Convolutional Deep Long Short-Term Memory (CDLSTM), inspired by Chen et al. [16]. The CDLSTM model uses combination of different layer's specific properties to classify occupancy state from raw data. Mainly convolutional operation (CD) is used for extracting features from raw data, while a DLSTM is used to understand the temporal dependence of the data.

The CDLSTM model functions in the following way. Windows of raw time-series data enter first the convolutional layer, which slides a filter window to extract the characteristics. Extracted features are then passed through the pooling operation, which compresses the features by removing less important local features. A Long Short-Term Memory (LSTM) belongs to Recurrent Neural Network (RNN), commonly used with sequential data processing, since RNNs are good in finding dependence of features over time sequence. LSTM are an improvement over regular RNN because of their ability to learn long-term dependencies. Chen et al. in their work used a Bidirectional LSTM (BLSTM), where the data flows from the past but also from the future through the model. In this work, unidirectional LSTM showed better performance than the proposed BLSTM. A Deep LSTM (DLSTM) model, consisting of more than one LSTM layer is used. After each LSTM layer, there is a dropout layer used for regularization.

Regularization is the name for a technique used to solve the overfitting problem. The problem of overfitting results in a model showing significantly better performance on training dataset then on testing, which is common with deep learning models. A dropout layer randomly drops out hidden nodes from the neural layer during the training time. In each loop, different nodes are dropped, making model to learn representations which are more robust to the noise in the data. While during the testing time, nodes are not dropped.

The following DLSTM layers in the framework of the neural model used in this research are fully connected dense layers, with their dropout layers as well. The fully connected part of the model is used to learn more abstract features in the data such as learning non-linear combinations of input features and preparing the features to their final representation. The final representation of the learned features from raw time-series data is then fed finally to the softmax classification layer. The softmax classification layer, translates given features into classes, which are in the case of this work occupancy states.

2.2 Proposed transfer method

Training a deep learning model such as the proposed CDLSTM requires significant amount of training data. Labelled training data is difficult to collect when it comes to room occupancy, especially the occupancy count or level. Therefore, in this work the transfer method of CDLSTM model is studied.

The principle behind transfer learning is that a model is trained on a labelled and extensive dataset, called source dataset. The acquired knowledge from training the source model can then be transferred to solve a similar problem on a target dataset. The weights of the trained CDLSTM model have then been transferred to a target CDLSTM model which was fine-tuned on a target dataset.

2.3 Data acquisition

Source dataset for the initial training of the CDLSTM model was collected using temperature and CO_2 IoT sensor in a meeting room in one office building located in Helsinki, Finland. This room is named Source Room. The measurements were done in period from February and March 2020. The source room is rather small with seats for four persons and with approximate size of 6 m^2. It is located in the middle of the floor and surrounded by an open office area. Ventilation is with constant airflow of 15 l/s consisting of 100% fresh outdoor air. The environmental IoT sensor was placed on the desk in the room. Camera for collecting ground truth occupancy was installed on the ceiling, just above the door.

Ground truth for the source model was acquired using video camera recording (the image was blurred for privacy purposes), which was used for manual counting of people in the source room. Target for the transfer of occupancy prediction model in this work is a large meeting room located in a hospital building in Finland. The Target Room has a capacity for 12 people with area of 21 m^2 and by the design, its ventilation system is designed to be of variable airflow. Temperature and CO_2 were collected with IoT sensor. The sensor was placed on the conference desk. Ground truth for transfer dataset has been acquired using infrared time of flight based people counting sensors. Automatic counting using IR-based sensors has a known problem with missing counts [21] and therefore this count was manually corrected using the presence and noise sensor in the room.

The measurement was done in two periods. First period, or period with constant airflow ventilation was from March and April 2021. Second period with variable airflow was from May to July 2021. During the measurement, the problem with the ventilation setting was noticed, the room did not operate with the expected variable airflow, because the settings were overridden to constant maximum design airflow of 90 l/s (first measurement period). When this was noticed, the setting has been changed to a predefined variable airflow (second measurement period).

Variable airflow in the room is controlled by the signal from three BMS sensors; PIR, CO_2 and temperature. The airflow rate is possible to control between 30 and 90 l/s. With variable airflow operation, it was also noticed that it is not performed as expected. In reality, the airflow almost never dropped under ~50 l/s, even though the room was empty for days at times. The source of the problem was too high room temperature compared to the setpoint (user adjusted between 18.5 and 23.5°C) which BMS tried to decrease by increasing the airflow. On the other hand, the air supply temperature was too high (about 21°C), making it impossible to cool down the room and therefore keeping the airflow at least on the medium level. Therefore, there is no clear connection between airflow and occupancy as such, making it more difficult to use the occupancy prediction model, which is addressed later in this paper, where the robustness of the transferred model is tested.

Therefore, in this work we have two periods with different ventilation strategies in same room, during March and April there was constant airflow, while from May onwards the work is performed with variable airflow data.

2.4 Experimental setup

Data was gathered from IoT temperature and CO_2 sensors with a sampling rate of one minute, while ground truth for the source dataset was acquired in three-minute intervals. Therefore, all other measurements were resampled to three minutes.

Before it was used for the deep learning model, the raw sensor data was preprocessed. Missing values, caused by connection problems or other IoT sensor problems, were interpolated using polynomial interpolation of second order. Raw sensor data was noisy and through the experimentation we have noticed that using data as such, accuracy of model was lower and with longer computational time. Therefore, we decided to smooth the data before the model input. For this purpose, smoothing based on Kalman filtering with Python package *tsmoothie* [22] was used. Kalman smoothing was performed on time-series level components.

During the initial experimentation, it was noticed that partly balancing the classes in the dataset also improved the results. Imbalanced classes in the dataset means that certain class classes are overrepresented then the other classes. In this case, the occupancy class of zero or empty room makes large majority of the dataset, while occupied state makes smaller portion of the dataset. Training the deep learning model on the imbalanced dataset can lead to a model with lower sensitivity to classes which are underrepresented. In this study, this issue was reduced by removing the periods when occupancy is expected to be zero, such as during the nighttime or the weekend.

The last thing to do before using the data with neural model is to normalize it. Z-score normalization was used, where the values are normalized according to their mean and standard deviation. Data was normalized using the mean and standard deviation from the training data, while for the later use of the model with variable airflow, a sliding method was used as described in section 3.3.

Normalized and preprocessed data was then ready to be used with CDLSTM model. The model was created using the deep learning Python package *Tensorflow/Keras* [23] and was designed as follows. The convolutional layer was created with 100 output filters, a kernel size of three which specifies the 1D convolutional window and pooling size of two, with ReLU activation function. Following is deep LSTM model, consisting of three LSTM layers, with hidden sizes of 100, 150 and 200 respectively. After each layer there is a dropout layer served for regularization with masking probability of 0.5. Following LSTM layers, there are two fully connected dense layers with hidden size 200 and 300. Another dropout layer is placed between the fully connected layers with a probability of 0.3. At the end, there is a softmax layer classifying the model outputs to classes, in our cases occupancy classes.

Another important setting for the CDLSTM model is the batch size and window sequence length and this varies depending on the sampling rate of the data on which the model was trained. For example, when using the dataset with three-minute sampling rate, a batch size of 16 and a sequence length of four were used. This means that one window was a size of four timesteps or twelve minutes which were fed to the model in batches of 16 windows at a time. The exact size of batches and sequences were found with the trial-and-error method.

2.5 Performance metric

To evaluate the performance of the model, the main metric used was the Matthews correlation coefficient (MCC). Similar work on occupancy estimation used a more traditional accuracy score or an F1 score, which happens to show dangerously optimistic results on overly imbalanced datasets. Since the rooms in question are majority of the time vacant, the number of times a vacant class (zero) is present in the data set is much higher compared to the number of times the room is occupied. Which means that if a model would always predict the room is vacant, it would be awarded with a high accuracy score. The MCC score, on the other hand, gives more understandable accuracy metrics for those cases: to achieve a high quality score, the classifier (such as the occupancy prediction model) has to make a prediction in majority of positive and negative cases independently of their ratios in the overall dataset [24].

On the other hand, MCC score gives a very general metric, which is not straightforward for the potential use cases of this work. To make the results more understandable additional performance metrics were developed. Aside from the room occupancy in each timestep, from the perspective of optimizing HVAC system and space utilization, it is useful to know the following about the meeting room occupancy; the time of the first and last occupancy in the day and the duration of room occupancy. Therefore, the corresponding metrics were developed. The difference in the first or last occupancy of the day between the measured and predicted occupancy is shown as averaged throughout the period and in minutes. Duration of room occupancy is shown as a percentage of the average day being occupied.

3. Results

3.1 Source model training

The source model for occupancy estimation was trained with data from source room. Training of the model was done using 22 days (14 days occupied) and validation of the training was done with the following three days. The model was first trained using the raw sensor data in 30 epochs and then in the second iteration smoothed data was used. Using raw and smoothed version of the same data showed better performance during training and it served as an additional regularization technique, making model less prone to overfitting. Trained source model has shown MCC score of 0.85.

3.2 Transfer to the target room

Transfer learning of the source model for successful occupancy detection is analyzed from the point of minimum training data needed from the target room.

Process for re-training of CDLSTM model considers which layers of the model are frozen and which are retrained with the new data and in which way. If a particular layer is not frozen, it can be trained from scratch (weights are reinitialized) or the layer weights can be transferred and then retrained.

Models compared in this work are Model which was trained using training data from only target dataset is called Baseline model, while model which was not retrained at all is Pure source model. The Transfer model is a model with pre-trained weights for all layers, which were retrained.

Goodness of the models is first assessed with MCC score and then the best cases are further analyzed using the additional performance metrics. In **Tab. 1** MCC score of different models with different training data length is presented. Analyzing the results from the training length perspective, it can be seen that with already two days of ground-truth data with transfer learning it is possible to get high MCC score, as with longer training period and nearly as

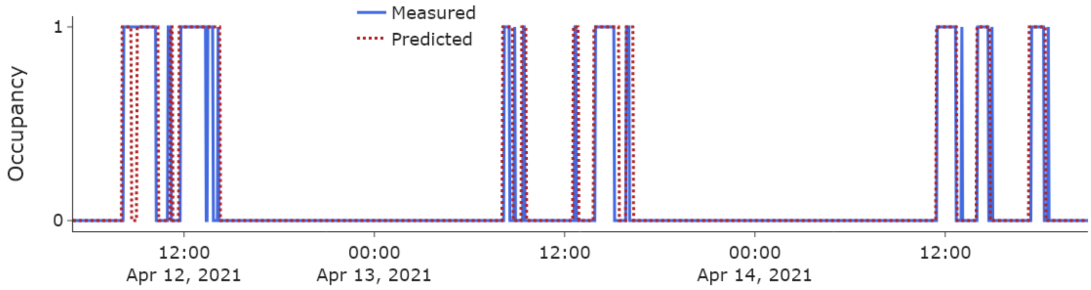

Fig. 1 - Three-day showcase of the selected target model performance and comparison with measured occupancy.

high as with source model. Without transfer method for two days training the best score would be 0.73, while for 5 days 0.80.

Tab. 1 – Comparison of different occupancy detection models performance with regards to the length of training data shown with MCC score

Model	1 day	2 day	3 day	5 day
Baseline model	0.67	0.73	0.79	0.80
Pure source model	0.71	0.71	0.71	0.71
Transfer model	0.77	0.82	0.82	0.82

Tab. 2 – Additional performance metrics for selected models of different training length and method, as daily average occupied time and average difference of first and last occupancy in a day

Model	MCC	Daily avg. occupied time [%]	First occ. avg. difference [min]	Last occ. avg. difference [min]
Ground truth	-	7.5	-	-
2 day - Baseline model	0.73	11	5	28
2 day - Transfer model	0.82	8.9	7	7
5 day – Baseline model	0.80	9.4	5	15

For further analysis, three interesting cases were chosen, which are made bold in **Tab. 1**. Selected cases are then further examined with additional performance metrics in order to understand what MCC score might mean for practical use and to select a final occupancy detection transfer model. This analysis can be seen in **Tab. 2**. Where the best result shows the case using two-day Transfer model. Having closest daily average occupied time to the ground truth of the shown cases and having only seven minutes average difference between the first and the last occupancy of the day. This case will be called the transferred occupancy detection model in the next sections. Transferred occupancy detection model's performance is visualized in **Fig. 1** where it is compared to the measured ground truth occupancy.

3.3 Robustness check - change in ventilation system operation

Robustness of ML models for predicting the room occupancy using environmental variables is important for using them commercially, as certain changes might happen in a building system operation. In this section the performance of the Transfer model is checked after a change in ventilation system operation occurs. In the target room, during the model transfer, ventilation operated with a constant airflow of about 90 l/s. Change in the operation was the demand-controlled variable air flow operation activation, where airflow was modulated between about 55 to 90 l/s, depending on the control logic described in section 2.3.

Robustness check on data from VAV operation was done using data measured during May 2021. During the process, it became clear that the way how z-score normalization is performed is important. In many time-series ML work the z-score normalization statistics derived from the training dataset is used also for the test data. This is suitable only for stationary time series, but not for non-stationary time series [25]. In this case introducing the variable airflow makes mean and standard deviation of CO_2 and temperature time series more susceptible to change over time. Therefore, a sliding window approach was used, where a normalization statistics was calculated on an arbitrarily chosen 14 days window prior to the day being predicted. More advanced approaches exist such as adaptive normalization [25], [26], but are out of the scope of this work.

Tab. 3 – Performance of previously transferred model after the room ventilation changed from CAV to VAV operation

Model	MCC	Daily avg. occupied time [%]	First occ. avg. difference [min]	Last occ. avg. difference [min]
Ground truth	-	5.7	-	-
Trans. model, norm. on training	0.71	8.4	-74	21
Trans. model, Sliding norm.	0.78	6.9	-27	2

Tab. 3 presents results from the robustness check of the transferred detection model with two normalization approaches. Results are presented using MCC for model accuracy and the additional performance metrics. The sliding window normalization produced significantly better results than using the normalization from the training phase. Using the sliding window normalization, detection performance of the model is close to the original performance of the detection model during ventilation operation in the CAV mode. Where largest difference in performance is in detection of the first daily occupancy.

4. Discussion

As mentioned in the literature review, plenty of studies were done exploring the accuracy of occupancy inferring methods from environmental variables. On the other hand, very little was done on scalability of those methods and therefore they have not found their way in the industry.

In this work, main obstacle was tried to be avoided or at least minimized by using transfer learning method. Having a previously trained model on large dataset and having a small amount of ground truth data (two days) from a different room, it is possible to get good occupancy detection accuracy.

Second obstacle, is what happens after certain conditions in the room change, how will the model perform? In this work, transferred model with the small amount of data from constant airflow conditions was tested after the conditions in room have changed to variable airflow. The test has shown that it is possible to keep the performance on same level, if normalization technique is improved.

For wider usage of similar models, probably even with this method, the problem is not solved. However, a further work should continue with removing the need for ground truth collection completely, if possible. Additionally, methods for monitoring drift of such models should be explored. Where the biggest issue is not having ground truth at all.

In this work, as a tool to increase robustness, a sliding version of z-score normalization was used. However, there are other state-of-the-art normalization methods for non-stationary time series, which should be explored.

Finally, occupancy detection is good, but much more valuable is knowing the number of people or at least the level of room occupancy. Minimizing the training data needed for this purpose is difficult, since machine learning models need to see enough examples of every class. This is an important issue for future research in this field.

5. Conclusion

The main goal of the presented study was to explore the scalability of deep learning-based method for inferring room occupancy information using environmental IoT sensor. Previous research has focused on testing different ML based methods in order to get as high accuracy as possible, but since it is difficult to collect labelled training data the methods were not used widely. In this study, a transfer learning method has been applied and it showed that it is possible to create occupancy detection model of good accuracy with only two days of ground truth data, instead of several weeks.

Furthermore, a robustness of the transferred model was tested in different airflow conditions and accuracy did not drop significantly when sliding normalization was used.

6. References

[1] European Union, "Directive 2018/844 of the European Parliament and of the Council of 30 May 2018 amending Directive 2010/31/EU on the energy performance of buildings and Directive 2012/27/EU on energy efficiency," *Off. J. Eur. Union*, p. L 156/75-L 156/91, 2018.

[2] UN Environment and International Energy Agency, "Towards a zero-emission, efficient, and resilient buildings and construction sector," 2017.

[3] L. Pérez-Lombard, J. Ortiz, and C. Pout, "A review on buildings energy consumption information," *Energy Build.*, vol. 40, no. 3, pp. 394–398, 2008, doi: 10.1016/j.enbuild.2007.03.007.

[4] J.-H. Choi, B. Yang, and C. W. Yu, "Artificial intelligence as an agent to transform research paradigms in building science and technology," *Indoor Built Environ.*, p. 1420326X2110176, Jun. 2021, doi: 10.1177/1420326X211017694.

[5] Z. Wang, J. Liu, Y. Zhang, H. Yuan, R. Zhang, and R. S. Srinivasan, “Practical issues in implementing machine-learning models for building energy efficiency: Moving beyond obstacles,” *Renew. Sustain. Energy Rev.*, vol. 143, no. February, p. 110929, 2021, doi: 10.1016/j.rser.2021.110929.
[6] M. Bedford, R. Harris, A. King, and A. Hawkeswood, “Occupier density study,” *Br. Counc. Off. BCO*, 2013.
[7] M. Naor, G. D. Pinto, A. I. Hakakian, and A. Jacobs, “The impact of COVID-19 on office space utilization and real-estate: a case study about teleworking in Israel as new normal,” *J. Facil. Manag.*, 2021, doi: 10.1108/JFM-12-2020-0096.
[8] Z. Chen, C. Jiang, and L. Xie, “Building occupancy estimation and detection: A review,” *Energy Build.*, vol. 169, pp. 260–270, 2018, doi: 10.1016/j.enbuild.2018.03.084.
[9] B. Dong *et al.*, “An information technology enabled sustainability test-bed (ITEST) for occupancy detection through an environmental sensing network,” *Energy Build.*, vol. 42, no. 7, pp. 1038–1046, 2010, doi: 10.1016/j.enbuild.2010.01.016.
[10] L. M. Candanedo, V. Feldheim, and D. Deramaix, “A methodology based on Hidden Markov Models for occupancy detection and a case study in a low energy residential building,” *Energy Build.*, vol. 148, pp. 327–341, 2017, doi: 10.1016/j.enbuild.2017.05.031.
[11] M. Amayri, A. Arora, S. Ploix, S. Bandhyopadyay, Q. D. Ngo, and V. R. Badarla, “Estimating occupancy in heterogeneous sensor environment,” *Energy Build.*, vol. 129, pp. 46–58, 2016, doi: 10.1016/j.enbuild.2016.07.026.
[12] Z. Chen, M. K. Masood, and Y. C. Soh, “A fusion framework for occupancy estimation in office buildings based on environmental sensor data,” *Energy Build.*, vol. 133, pp. 790–798, 2016, doi: 10.1016/j.enbuild.2016.10.030.
[13] M. S. Zuraimi, A. Pantazaras, K. A. Chaturvedi, J. J. Yang, K. W. Tham, and S. E. Lee, “Predicting occupancy counts using physical and statistical Co2-based modeling methodologies,” *Build. Environ.*, vol. 123, pp. 517–528, 2017, doi: 10.1016/j.buildenv.2017.07.027.
[14] M. K. Masood, Y. C. Soh, and V. W. C. Chang, “Real-time occupancy estimation using environmental parameters,” *Proc. Int. Jt. Conf. Neural Netw.*, vol. 2015-Septe, pp. 1–8, 2015, doi: 10.1109/IJCNN.2015.7280781.
[15] M. K. Masood, C. Jiang, and Y. C. Soh, “A novel feature selection framework with hybrid feature-scaled extreme learning machine (HFS-ELM) for indoor occupancy estimation,” *Energy Build.*, vol. 158, pp. 1139–1151, 2018, doi: 10.1016/j.enbuild.2017.08.087.
[16] Z. Chen, R. Zhao, Q. Zhu, M. K. Masood, Y. C. Soh, and K. Mao, “Building occupancy estimation with environmental sensors via CDBLSTM,” *IEEE Trans. Ind. Electron.*, vol. 64, no. 12, pp. 9549–9559, Dec. 2017, doi: 10.1109/TIE.2017.2711530.
[17] B. Dong *et al.*, “Sensing and data acquisition,” in *Exploring occupant behavior in buildings: Methods and challenges*, Springer International Publishing, 2017, pp. 77–105.
[18] B. Dong, D. Yan, Z. Li, Y. Jin, X. Feng, and H. Fontenot, “Modeling occupancy and behavior for better building design and operation—A critical review,” *Build. Simul.*, vol. 11, no. 5, pp. 899–921, 2018, doi: 10.1007/s12273-018-0452-x.
[19] S. J. Pan and Q. Yang, “A survey on transfer learning,” *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010, doi: 10.1109/TKDE.2009.191.
[20] M. Weber, C. Doblander, and P. Mandl, “Towards the detection of building occupancy with synthetic environmental data,” 2020, [Online]. Available: http://arxiv.org/abs/2010.04209
[21] J. Kuutti, K. H. Blomqvist, and R. E. Sepponen, “Performance of commercial over-head camera sensors in recognizing patterns of two and three persons : A case study,” 2013.
[22] M. Cerliani, “tsmoothie.” 2020. [Online]. Available: https://github.com/cerlymarco/tsmoothie
[23] F. Chollet and others, “Keras.” 2015.
[24] D. Chicco and G. Jurman, “The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation,” *BMC Genomics*, vol. 21, no. 1, p. 6, Jan. 2020, doi: 10.1186/s12864-019-6413-7.
[25] E. Ogasawara, L. C. Martinez, D. De Oliveira, G. Zimbrão, G. L. Pappa, and M. Mattoso, “Adaptive normalization: A novel data normalization approach for non-stationary time series,” in *The 2010 International Joint Conference on Neural Networks (IJCNN)*, 2010, pp. 1–8.
[26] N. Passalis, A. Tefas, J. Kanniainen, M. Gabbouj, and A. Iosifidis, “Deep Adaptive Input Normalization for Time Series Forecasting,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 9, pp. 3760–3765, Sep. 2020, doi: 10.1109/TNNLS.2019.2944933.